MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 2, 2016/2017

DPJ5018 – PROGRAMMING IN JAVA

(For Diploma students only)

24 FEBRUARY 2017
9.00 a.m - 11.00 a.m
(2 Hours)

INSTRUCTIONS TO STUDENT

1. This question paper consists of **16 pages** with **3 sections**.

   Section A: Multiple Choice Questions (10 Marks)
   Section B: Structured Questions (60 Marks)
   Section C: Application Program (30 Marks) ***(Attempt ONE (1) question only)***

2. Write **ALL** your answers in the Answer Booklet provided.

SECTION A: MULTIPLE CHOICE QUESTIONS (10 MARKS)

Instructions: Answer ALL questions. Write your answers in the Answer Booklet provided.

1. Which of the following is the component class?

   A. JFrame
   B. JApplet
   C. JTextArea
   D. None of the above

> *"A special kind of Java Programs*
> *that can run directly from Java-compatible Web browser."*

2. Which of the following **BEST** describes the above statement?

   A. Java SE
   B. Java IDE
   C. Java Applet
   D. Java Application

3. An object must be an instance of `ActionListener` in order to be a listener of _____.

   A. ItemEvent
   B. EventObject
   C. ActionEvent
   D. actionPerformed

4. What code would you use to construct a 24 Point bold serif font?

```
A. new Font("SERIF", 24, BOLD");
B. new Font("SERIF", Font.BOLD,24);
C. new Font("BOLD ", 24,Font.SERIF);
D. new Font(Font.SERIF, "BOLD", 24)
```

5. Which of the following statement is **CORRECT** to draw a circle with radius 10; align centered at (20, 20)?

```
A. g.drawOval(20, 20, 20, 20)
B. g.drawOval(30, 30, 40, 40)
C. g.drawCircle(20, 20, 40, 40)
D. g.drawCircle(20, 20, 20, 20)
```

Continued...

6. Which of the following is a **VALID** syntax of creating a two-dimensional array with three rows and four columns?

A. `Student = new int [3] [4];`
B. `int [] [] Student =new int [4];`
C. `int [] [] Student = new int [4] [3];`
D. `int [] [] Student = new int [3] [4];`

7. Evaluate the value of x in the expression given below:

$$x = 6 - 2 + 10 \,\%\, 4 + 7$$

A. 10
B. 12
C. 13
D. 14

8. What is the output of the following program?

```
1  class Number
2  {  int a;
3
4     void display ( )
5     {
6        System.out.println(a);
7     }
8  }
9
10 class Numb extends Number
11 {  int b;
12
13    void display ( )
14    {
15       System.out.println(b);
16    }
17 }
18
19 public class MainClass
20 {
21    public static void main(String[] args)
22    {
23       Numb N = new Numb ( );
24       N.a = 3;
25       N.b = 4;
26       N.display ( );
27    }
28 }
```

A. 0
B. 3
C. 4
D. Compilation Error

Continued...

9. The `static` field of a class in Java is having all these characteristics **EXCEPT**:

   A. Not tied to a specific object.
   B. Not shared among objects of the class.
   C. Use keyword static in declaring the variable.
   D. Affected if one of the objects changes the value of a static field.

10. Which of the following keywords is not use in the exceptional handling approach?

   A. Try
   B. Catch
   C. Finally
   D. Thrown

Continued...

SECTION B: STRUCTURED QUESTION (60 MARKS)

Instructions: Answer ALL questions. Write your answers in the Answer Booklet provided.

QUESTION 1 [20 MARKS]

a) Given these values:

```
int p=12, q=6, r=9, a;
```

- Evaluate the following expressions.
- Show your workings clearly.

*** NOTE:** *The expressions are* ***NOT*** *related to each other.*

i) `a = ++p % 4 + q * r++` [2 Marks]

ii) `a = r-- / (q + 3) - 5 % 3` [2 Marks]

b) Write a static method named `CalcTrapezoidVol` that calculate the volume of a trapezoid prism, using the following formula:

$$\text{Trapezoid Prism} = \left(\frac{\text{top length} + \text{bottom length}}{2}\right) \times height \times length$$

This method has four `double` parameters, which are:

- `a`: the length of the top,
- `b`: the length of the bottom,
- `h`: the height of the prism,
- `l`: the length of the prism

This method will return a `double` value which is the volume of the trapezoid volume.

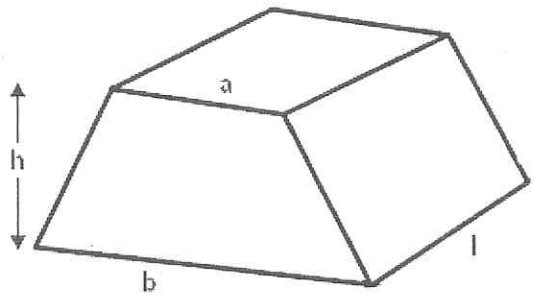

Figure 1: The Trapezoid Prism

[5 marks]

Continued...

c) `CakeQuantity` is an `int` array type that stores the quantity of cupcakes for the four type of flavours. While `CakePack` is a `String` array that used to store the type of pack for these four flavours.

```
int [ ] CakeQuantity = {5, 25, 15, 10};
String [ ] CakePack = new String [4];
```

Write a loop statement to determine the type of pack for the quantity of the cupcakes based on *Table 1*, and store the value into the array `CakePack`.

Quantity	Pack
16-30	PartyPack
11-15	HappyPack
6-10	ValuePack
Below 6	MiniPack

Table 1

[7 marks]

d) Write Java code to perform the following questions based on the given array created:

```
int [] collection = new int [5];
```

i) To allow the user to key in the values for the array collection. [2 marks]

ii) To calculate the total of the values of collection. [2 marks]

QUESTION 2 [20 MARKS]

a) Write one class based on the UML class diagram in Figure 2. [3 marks]

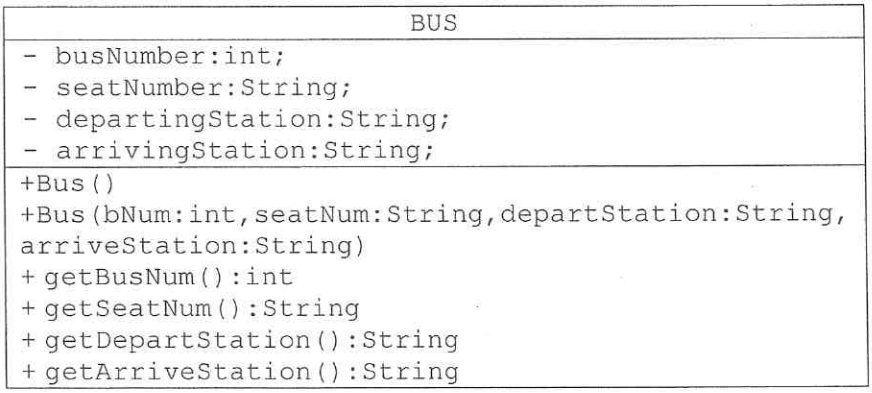

Figure 2: UML Diagram

Continued...

b) Determine the output of the following program. [12 marks]

```
1  class Vehicle
2  {
3    public void move ()
4      {
5       System.out.println ("Vehicles are used for moving from
6  one place to another ");
7      }
8  }
9  class Car extends Vehicle
10 {
11     public void move ()
12     {
13      super.move ();
14      System.out.println ("Car is a good medium of transport
15 ");
16      System.out.println ("Car has at least two doors");
17     }
18 }
19 class Motorcycle extends Vehicle
20 {
21     public void move ()
22     {
23      super.move ();
24      System.out.println ("Motorcycle is one of the medium of
25 transport ");
26      System.out.println ("Motorcycle has at least two tyres");
27     }
28 }
29 class Lorry extends Vehicle
30 {
31     public void move ()
32     {
33      System.out.println ("Lorry is a transport used for carry
34 big loads ");
35      System.out.println ("Maximum speed limit for lorries is
36 90km/h");
37     }
38 }
39 public class TestCar
40 {
41     public static void main (String args [])
42     {
43         Vehicle b = new Car();
44         Vehicle c = new Motorcycle();
45         Vehicle d = new Lorry();
46         c.move ();
47         System.out.println("");
48         d.move();
49         System.out.println("");
50         b.move();
51         System.out.println("");
52     }}
53
```

Continued...

c) Determine the output for the following program. [5 marks]

```
1  public static void first()
2  {
3  System.out.println("START EXECUTE.");
4
5  try
6  {  acer(); }
7
8  catch (Exception e)
9  { System.out.println("Oh! I'm being caught.");
10   System.out.println(e.getMessage()); }
11
12 System.out.println("RUN!");
13 }
14
15 public static void acer()
16 { System.out.println("Acer is ON.");
17
18 try
19 {  dell(); }
20
21 catch (ArithmeticException e)
22 { System.out.println("Caught at second");
23   System.out.println(e.getMessage());  }
24
25 System.out.println("Ending second.");
26 }
27
28 public static void dell()
29 { System.out.println("I'm Dell.");
30   throw new RuntimeException("unknown error");
31   System.out.println("Ending third."); }
```

Continued...

QUESTION 3 [20 MARKS]

a) Write Java code to create a frame as shown in *Figure 3*. The frame should be appeared and exits the program when user closes the frame.

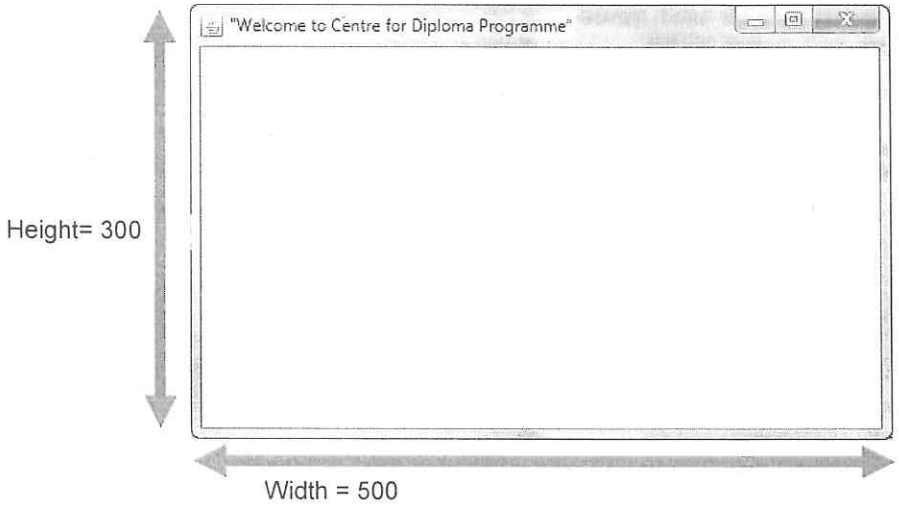

Figure 3

[5 marks]

b) Write Java code to create six buttons arranged in a frame as shown in *Figure 4*.

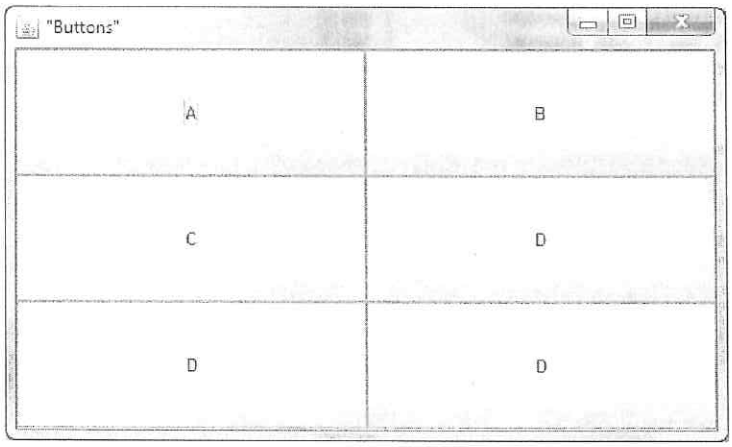

Figure 4

[4 marks]

Continued...

c) Write a Java code to create three checkbox with the following properties. Refer to the sample output shown in *Figure 5*.

- Grid Layout with 3 rows and 1 column
- Font colour : RED
- Selected State: February

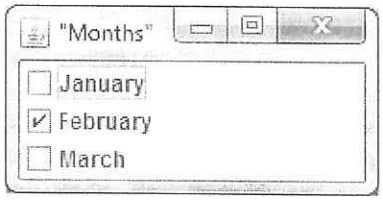

Figure 5

[5 marks]

d) Write a Java code to draw two arcs with the following properties:
Arc1 properties:

- Outline colour : RED
- Upper-left Corner where x is 150, y is 100
- Height : 100
- Width : 50
- Starting angle : 10
- Ending angle : 90

Arc 2 properties:

- Fill colour : YELLOW
- Upper-left Corner where x is 100, y is 200
- Height : 100
- Width : 100
- Starting angle : 0
- Ending angle : 90

[6 marks]

Continued...

SECTION C: FULL PROGRAM (30 MARKS)

*Instructions: Answer **ONE (1)** question only. Write your answers in the Answer Booklet provided.*

QUESTION 1 [30 MARKS]

Perform the following instructions by referring to the UML diagram of the classes as shown in *Figure 6* and the sample output as shown in *Figure 7*.

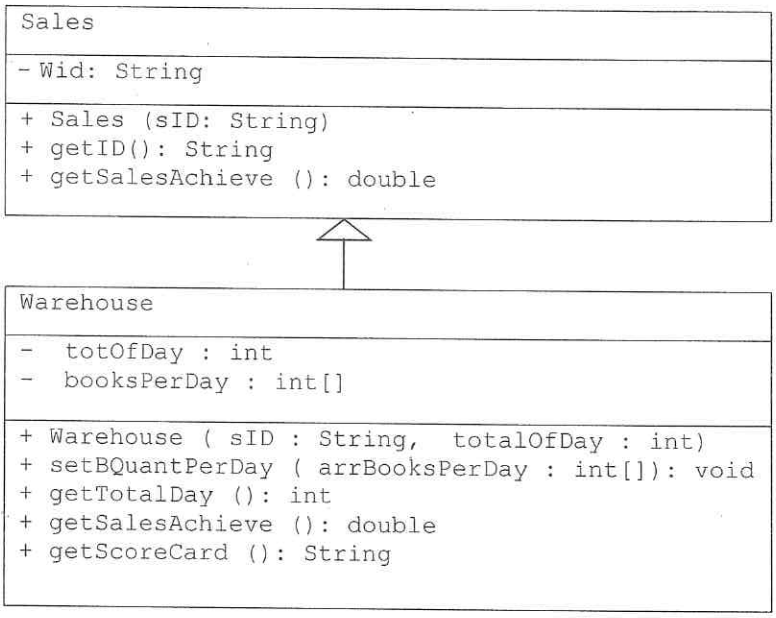
Sales
- Wid: String
+ Sales (sID: String)
+ getID(): String
+ getSalesAchieve (): double
Warehouse
- totOfDay : int
- booksPerDay : int[]
+ Warehouse (sID : String, totalOfDay : int)
+ setBQuantPerDay (arrBooksPerDay : int[]): void
+ getTotalDay (): int
+ getSalesAchieve (): double
+ getScoreCard (): String

Figure 6: UML Diagram

```
run:
Enter Warehouse Sales ID: BigBadWolf2016
How Many Day of Sales Recorded: 4
Sales Recorded on Day 1 (Books Quantity): 500
Sales Recorded on Day 2 (Books Quantity): 300
Sales Recorded on Day 3 (Books Quantity): 600
Sales Recorded on Day 4 (Books Quantity): 450

 The Record of Warehouse Sales Performance
-------------------------------------------
Warehouse Sales ID: BigBadWolf2016
Sales Recorded for: 4 day
Sales Achieved: RM 30000.00
Score Card: Awesome
BUILD SUCCESSFUL (total time: 1 minute 23 seconds)
```

Figure 7: Sample Output

Continued...

Write a class named `Sales` which consists of the following data field and methods:

- Data field:
  - `Wid`: Stores the warehouse sales ID as private.

- The `Sales (...)` constructor accepts the warehouse sales ID as an argument and assigns it to the appropriate data field.

- Methods:
  - `getID()`: Returns the warehouse sales ID.
  - `getSalesAchieve ()`: Returns the daily base sales target which is RM 3000.

Write a class named `Warehouse` which is a subclass of Sales. The Warehouse class consists of the following data fields and methods:

- Data fields (as private):
  - `totOfDay`: Stores the total of day selected for the warehouse sales tracking record.
  - `booksPerDay`: An int type of array that stores the quantity of books sold per day.

- The `Warehouse (...)` constructor that accepts two-argument: the warehouse sales ID and the total of day selected for the warehouse sales tracking record, and initialize those arguments with appropriate data fields accordingly.

- Methods:
  - `setBQuantPerDay (...)`: Define appropriate argument to represent `booksPerDay` data field and initialize it accordingly.

  - `getTotalDay ()`: Returns the total of day selected for the warehouse sales tracking record.

  - `getSalesAchieve ()`: Overrides the superclass's method to calculate and return the sales achieved for the said total of day. The books sales achievement is based by the total books sold in the days selected. The books per day array used for the quantity of books sold for each said day. If the books sold more than 500 books in a selected group of days, the sales score is calculated based on the total books sold for each day and multiply with the bonus sales of a day rate RM 40. Otherwise, the sales score is calculated as RM 2000 by default.

  - `getScoreCard ()`: Returns the sales Score Card status. The score card is rated as "Awesome" if the sales achieve more than RM 3100 in the days selected. Otherwise, the score card is rated as "Insufficient".

Continued...

Write a class named `TestWarehouseSales` to test the `Warehouse` class.

- Prompt the user to enter the Warehouse Sales ID and the total days of the sales performance to be recorded.
- Create an object of Warehouse by passing the Warehouse Sales ID and the total days of the sales selected as entered by user.
- Create an array to store the quantity of books. The array size is based on the total days of the sales selected.
- Prompt the user to enter the total of books sold for each day and store each value into the array.
- Call the `setBQuantPerDay (...)` method and pass the array to the method.
- Display the Warehouse Sales Performance record which includes the following details by calling the appropriate methods:

    - Warehouse Sales ID
    - The total of day selected for the warehouse sales tracking record
    - Sales Achieved for the selected days
    - The sales - Score Card status

[30 marks]

Continued...

QUESTION 2 [30 MARKS]

a) Write a Java program that works as a simple calculator name as `SimpleCalc.java`. Use a grid layout to arrange a title, two text fields for the user to enter digits and four buttons for the ADD, SUBTRACT, MULTIPLY and DIVIDE operations. Add a text field to display the result.

When any of the operation button is clicked, the program will calculate and displays the result. A sample run of the program is shown in *Figure 8* and *Figure 9*.

Refer to the following details and *Figure 10* for the user interface:

- The size of the frame is 500 x 300.
- Background Colour : Orange

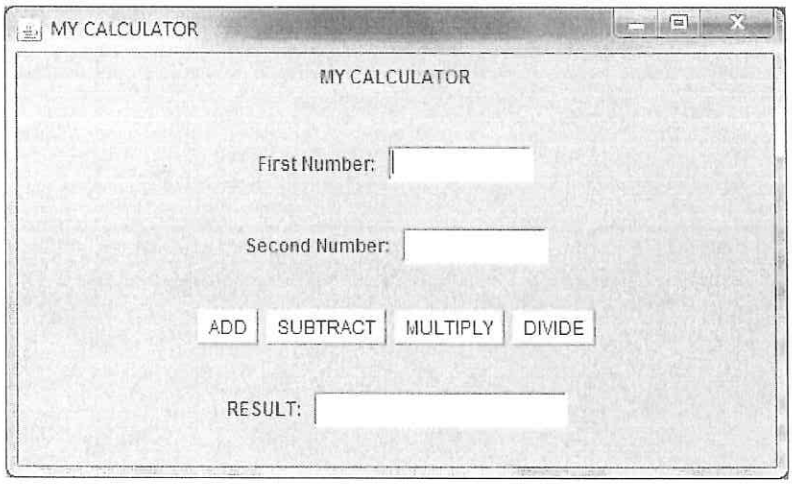

Figure 8: Sample Output

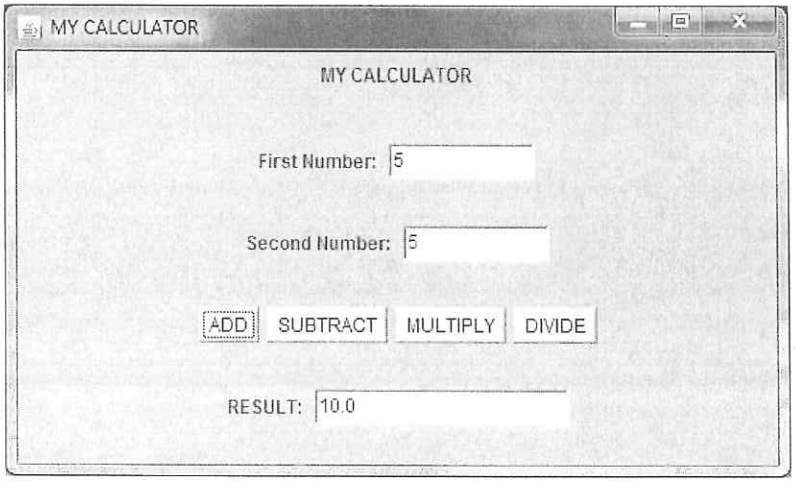

Figure 9: Sample Output with Result

Continued...

PANEL 1 "**MY CALCULATOR**" label center align
PANEL 2 "**First Number**" label "Number1" text field
PANEL 3 "**Second Number**" label "Number2" text field
PANEL 4 "**ADD**" button "**SUBTRACT**" button "**MULTIPLY**" button "**DIVIDE**" button
PANEL 5 "**RESULT**" label "Result" text field

Figure 10: User Interface

[20 marks]

Continued...

b) Convert the following program from a Java application into an Applet. [8 marks]

```
1  import java.awt.*;
2  import java.awt.event.*;
3
4  public class BalloonFrame extends Frame implements
5  ActionListener{
6
7     private Button grow, shrink, left, right, close;
8     private Balloon theBalloon;
9
10    public BalloonFrame() {
11       setTitle("Balloon");
12
13       setLayout(new FlowLayout());
14
15       grow = new Button("Grow");
16       add(grow);
17       grow.addActionListener(this);
18
19       shrink = new Button("Shrink");
20       add(shrink);
21       shrink.addActionListener(this);
22
23       left = new Button("Left");
24       add(left);
25       left.addActionListener(this);
26
27       right = new Button("Right");
28       add(right);
29       right.addActionListener(this);
30
31       close = new Button("Exit");
32       add(close);
33       close.addActionListener(this);
34
35       theBalloon = new Balloon();
36    }
37
38    public static void main(String args[]) {
39       BalloonFrame f = new BalloonFrame();
40       f.setSize(500, 500);
41       f.setVisible(true);
42    }
43
44    public void actionPerformed(ActionEvent event) {
45       if (event.getSource() == grow)
46          theBalloon.grow();
47       if (event.getSource()  == shrink)
48          theBalloon.shrink();
49       if (event.getSource()  == left)
50          theBalloon.moveLeft();
51       if (event.getSource() == right)
52          theBalloon.moveRight();
53       if (event.getSource() == close)
54          System.exit(0);
55
56       repaint();
57    }
58    public void paint(Graphics g) {
59       theBalloon.display(g);   } }
```

Continued...

c) Create an HTML file to invoke the Applet you have created in *Question 2 b)* with the Applet size; width is 400 and height is 200.

[2 marks]

End of Page.